

FFFFFFFFFFFFFFFF	111	111	AAAAAAAAAA	
FFFFFFFFFFFFFFFF	111	111	AAAAAAAAAA	
FFFFFFFFFFFFFFFF	111	111	AAAAAAAAAA	
FFF	111111	111111	AAA	AAA
FFF	111111	111111	AAA	AAA
FFF	111111	111111	AAA	AAA
FFF	111	111	AAA	AAA
FFF	111	111	AAA	AAA
FFF	111	111	AAA	AAA
FFFFFFFFFFFFFF	111	111	AAA	AAA
FFFFFFFFFFFFFF	111	111	AAA	AAA
FFFFFFFFFFFFFF	111	111	AAA	AAA
FFF	111	111	AAAAAAAAAAAAAAAA	AAA
FFF	111	111	AAAAAAAAAAAAAAAA	AAA
FFF	111	111	AAAAAAAAAAAAAAAA	AAA
FFF	111	111	AAA	AAA
FFF	111	111	AAA	AAA
FFF	111	111	AAA	AAA
FFF	111	111	AAA	AAA
FFF	1111111111	1111111111	AAA	AAA
FFF	1111111111	1111111111	AAA	AAA
FFF	1111111111	1111111111	AAA	AAA

```
CCCCCCCC  RRRRRRRR  EEEEEEEEE  FFFFFFFF  CCCCCCCC  BBBB88888
CCCCCCCC  RRRRRRRR  EEEEEEEEE  FFFFFFFF  CCCCCCCC  BBBB88888
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CC         RRRRRRRR  EEEEEEEEE  FFFFFFFF  CC         BBBB88888
CC         RRRRRRRR  EEEEEEEEE  FFFFFFFF  CC         BBBB88888
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CC         RR      RR      FF         CC         BB      BB
CCCCCCCC  RR      RR      FF         CCCCCCCC  BBBB88888
CCCCCCCC  RR      RR      FF         CCCCCCCC  BBBB88888
```

```
LL         IIIIII  SSSSSSSS
LL         IIIIII  SSSSSSSS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SSSSSS
LL         II      SSSSSS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SS
LLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLL IIIIII  SSSSSSSS
```

```
0001 0 MODULE CREFCB (  
0002 0     LANGUAGE (BLISS32),  
0003 0     IDENT = 'V04-000'  
0004 0 ) =  
0005 1 BEGIN  
0006 1  
0007 1  
0008 1 *****  
0009 1 *  
0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0012 1 *  ALL RIGHTS RESERVED.  
0013 1 *  
0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0019 1 *  TRANSFERRED.  
0020 1 *  
0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0023 1 *  CORPORATION.  
0024 1 *  
0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0027 1 *  
0028 1 *  
0029 1 *****  
0030 1  
0031 1 ++  
0032 1  
0033 1 FACILITY: F11ACP Structure Level 1  
0034 1  
0035 1 ABSTRACT:  
0036 1  
0037 1     These routines create and initialize a file control block  
0038 1     from the given file header.  
0039 1  
0040 1 ENVIRONMENT:  
0041 1  
0042 1     STARLET operating system, including privileged system services  
0043 1     and internal exec routines. These routines must be called in  
0044 1     kernel mode.  
0045 1  
0046 1  
0047 1 --  
0048 1  
0049 1  
0050 1 AUTHOR: Andrew C. Goldstein, CREATION DATE: 14-Dec-1976 16:48  
0051 1  
0052 1 MODIFIED BY:  
0053 1  
0054 1     A0100   ACG0001   Andrew C. Goldstein, 10-Oct-1978 20:01  
0055 1     Previous revision history moved to F11A.REV  
0056 1  
0057 1 **
```



CREFCB  
V04-000

C 14  
16-Sep-1984 00:54:07  
14-Sep-1984 12:29:25

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[F11A.SRC]CREFCB.B32;1 Page 2 (1)

```
.. 58      0058 1
.. 59      0059 1
.. 60      0060 1 LIBRARY 'SYSS$LIBRARY:LIB.L32';
.. 61      0061 1 REQUIRE 'SRC$:FCPDEF.B32';
.. 62      0376 1
.. 63      0377 1
.. 64      0378 1 FORWARD ROUTINE
.. 65      0379 1 CREATE_FCB,
.. 66      0380 1 UPDATE_FCB      : NOVALUE;
```

```
68 0381 1 GLOBAL ROUTINE CREATE_FCB (HEADER) =
69 0382 1
70 0383 1 ++
71 0384 1
72 0385 1 FUNCTIONAL DESCRIPTION:
73 0386 1
74 0387 1 This routine creates an FCB and initializes it according to
75 0388 1 the given file header.
76 0389 1
77 0390 1 CALLING SEQUENCE:
78 0391 1 CREATE_FCB (ARG1)
79 0392 1
80 0393 1 INPUT PARAMETERS:
81 0394 1 ARG1: address of file header
82 0395 1
83 0396 1 IMPLICIT INPUTS:
84 0397 1 NONE
85 0398 1
86 0399 1 OUTPUT PARAMETERS:
87 0400 1 NONE
88 0401 1
89 0402 1 IMPLICIT OUTPUTS:
90 0403 1 NONE
91 0404 1
92 0405 1 ROUTINE VALUE:
93 0406 1 ADDRESS OF FCB
94 0407 1
95 0408 1 SIDE EFFECTS:
96 0409 1 FCB created and initialized
97 0410 1
98 0411 1 --
99 0412 1
100 0413 2 BEGIN
101 0414 2
102 0415 2 MAP
103 0416 2 HEADER : REF BBLOCK; ! file header argument
104 0417 2
105 0418 2 LOCAL
106 0419 2 FCB : REF BBLOCK; ! address of FCB created
107 0420 2
108 0421 2 EXTERNAL ROUTINE
109 0422 2 ALLOCATE, ! allocate dynamic memory
110 0423 2 INIT_FCB; ! initialize contents of FCB
111 0424 2
112 0425 2 ! Allocate an FCB sized and typed block. Then use the common routine
113 0426 2 ! to init it.
114 0427 2
115 0428 2
116 0429 2 FCB = ALLOCATE (FCB$C_LENGTH, FCB_TYPE);
117 0430 2 FCB[FCB$L_WLFL] = FCB[FCB$L_WLFL]; ! init null window list
118 0431 2 FCB[FCB$L_WLBL] = FCB[FCB$L_WLFL];
119 0432 2 FCB[FCB$L_STVBN] = 1; ! init start VBN to 1
120 0433 2 INIT_FCB (FCB, .HEADER);
121 0434 2 RETURN .FCB;
122 0435 2
123 0436 1 END; ! end of routine CREATE_FCB
```

CREFCB  
V04-000

E 14  
16-Sep-1984 00:54:07  
14-Sep-1984 12:29:25

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[F11A.SRC]CREFCB.B32;1  
Page 4  
(2)

				0004 00000
	7E	B4	7E	D4 00002
0000G	CF		8F	9A 00004
	52		02	FB 00008
10	A2	10	50	D0 0000D
14	A2	10	A2	9E 00010
2C	A2		A2	9E 00015
		04	01	D0 0001A
			AC	DD 0001E
0000G	CF		52	DD 00021
	50		02	FB 00023
			52	D0 00028
			04	0002B

.TITLE	CREFCB	
.IDENT	\V04-000\	
.EXTRN	ALLOCATE, INIT_FCB	
.PSECT	\$CODE\$,NOWRT,2	
.ENTRY	CREATE_FCB, Save R2	: 0381
CLRL	-(SP)	: 0429
MOVZBL	#180, -(SP)	
CALLS	#2, ALLOCATE	
MOVL	R0, FCB	
MOVAB	16(FCB), 16(FCB)	: 0430
MOVAB	16(FCB), 20(FCB)	: 0431
MOVL	#1, 44(FCB)	: 0432
PUSHL	HEADER	: 0433
PUSHL	FCB	
CALLS	#2, INIT_FCB	: 0434
MOVL	FCB, R0	: 0436
RET		

; Routine Size: 44 bytes,      Routine Base: \$CODE\$ + 0000



```

125 0437 1 GLOBAL ROUTINE UPDATE_FCB (HEADER) : NOVALUE =
126 0438 1
127 0439 1 ++
128 0440 1
129 0441 1 FUNCTIONAL DESCRIPTION:
130 0442 1
131 0443 1 This routine updates the file attributes of the file's primary FCB,
132 0444 1 if any, with the file attributes of the given header. The file size
133 0445 1 is preserved.
134 0446 1
135 0447 1
136 0448 1 CALLING SEQUENCE:
137 0449 1 UPDATE_FCB (ARG1)
138 0450 1
139 0451 1 INPUT PARAMETERS:
140 0452 1 ARG1: address of file header
141 0453 1
142 0454 1 IMPLICIT INPUTS:
143 0455 1 NONE
144 0456 1
145 0457 1 OUTPUT PARAMETERS:
146 0458 1 NONE
147 0459 1
148 0460 1 IMPLICIT OUTPUTS:
149 0461 1 PRIMARY_FCB: address of file FCB or 0
150 0462 1
151 0463 1 ROUTINE VALUE:
152 0464 1 NONE
153 0465 1
154 0466 1 SIDE EFFECTS:
155 0467 1 FCB is updated if it exists
156 0468 1
157 0469 1 --
158 0470 1
159 0471 2 BEGIN
160 0472 2
161 0473 2 MAP
162 0474 2 HEADER : REF BBLOCK; ! file header arg
163 0475 2
164 0476 2 LOCAL
165 0477 2 FCB : REF BBLOCK, ! local pointer to FCB
166 0478 2 MAP_AREA : REF BBLOCK, ! pointer to header map area
167 0479 2 MAP_POINTER : REF BBLOCK; ! pointer to scan map
168 0480 2
169 0481 2 EXTERNAL
170 0482 2 PRIMARY_FCB : REF BBLOCK, ! FCB of file in process
171 0483 2 HEADER_LBN; ! LBN of file header
172 0484 2
173 0485 2
174 0486 2 FCB = .PRIMARY_FCB;
175 0487 2 IF .FCB EQL 0 THEN RETURN;
176 0488 2
177 0489 2
178 0490 2 ! Get the known constants and the simple stuff from the file header
179 0491 2 (i.e., header LBN, file ID, starting VBN, file owner and file protection).
180 0492 2
181 0493 2

```

```

182 0494 2 FCB[FCBSL_HDLBN] = .HEADER_LBN;
183 0495 2 FCB[FCBSW_FID_NUM] = .HEADER[FH1SW_FID_NUM];
184 0496 2 FCB[FCBSW_FID_SEQ] = .HEADER[FH1SW_FID_SEQ];
185 0497 2 FCB[FCBSW_UICMEMBER] = .HEADER[FH1SB_UICMEMBER];
186 0498 2 FCB[FCBSW_UICGROUP] = .HEADER[FH1SB_UICGROUP];
187 0499 2 FCB[FCBSW_FILEPROT] = .HEADER[FH1SW_FILEPROT];
188 0500 2 IF .HEADER[FH1SV_SPOOL] THEN FCB[FCBSV_SPOOL] = 1;
189 0501 2 FCB[FCBSL_EFBLK] = ROT (.BBLOCK[HEADER[FH1SW_RECATTR], FAT$L_EFBLK], 16);
190 0502 2 IF .FCB[FCBSL_EFBLK] NEQ 0
191 0503 2 AND .BBLOCK[HEADER[FH1SW_RECATTR], FAT$W_FFBYTE] EQL 0
192 0504 2 THEN FCB[FCBSL_EFBLK] = .FCB[FCBSL_EFBLK] - 1;
193 0505 2
194 0506 2 ! Now scan the map area. Get the starting LBN if the f 2 is contiguous.
195 0507 2 !
196 0508 2
197 0509 2 MAP_AREA = .HEADER + .HEADER[FH1SB_MPOFFSET]*2;
198 0510 2 MAP_POINTER = .MAP_AREA + FM1$C_POINTERS;
199 0511 2 FCB[FCBSW_SEGN] = .MAP_AREA[FM1$B_EX_SEGNUM];
200 0512 2
201 0513 2 FCB[FCBSL_STLBN] = 0; ! assume non-contiguous file
202 0514 2 IF .HEADER[FH1SV_CONTIG]
203 0515 2 THEN
204 0516 2 BEGIN
205 0517 2 FCB[FCBSL_STLBN] = .MAP_POINTER[FM1$W_LOWLBN]; ! get low order LBN
206 0518 2 (FCB[FCBSL_STLBN]<16,85) = .MAP_POINTER[FM1$B_HIGHLBN]; ! and high order
207 0519 2 END;
208 0520 2
209 0521 2
210 0522 2 IF .FCB[FCBSL_EFBLK] GTR .FCB[FCBSL_FILESIZE]
211 0523 2 THEN FCB[FCBSL_EFBLK] = .FCB[FCBSL_FILESIZE];
212 0524 2
213 0525 1 END; ! end of routine UPDATE_FCB

```

```
.EXTRN  PRIMARY_FCB, HEADER_LBN
```

PC	Instruction	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418
----	-------------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------



2A	A2		61	9B	0004E	MOVZBW	(MAP_AREA), 42(FCB)	: 0511
		30	A2	D4	00052	CLRL	48(FCB)	: 0513
		0C	A3	95	00055	TSTB	12(R3)	: 0514
			09	18	00058	BGEQ	3\$	: 0517
30	A2	02	A0	3C	0005A	MOVZWL	2(MAP_POINTER), 48(FCB)	: 0518
32	A2		60	90	0005F	MOVB	(MAP_POINTER), 50(FCB)	: 0522
38	A2		64	D1	00063	CMPL	(R4), 56(FCB)	: 0523
			04	15	00067	BLEQ	4\$	: 0525
	64	38	A2	D0	00069	MOVL	56(FCB), (R4)	: 0525
			04	0006D	4\$:	RET		

; Routine Size: 110 bytes, Routine Base: \$CODE\$ + 002C

: 214 0526 1  
: 215 0527 1 END  
: 216 0528 0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	154	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPI,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	31	0	1000	00:01.9

COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:CREFCB/OBJ=OBJ\$:CREFCB MSRC\$:CREFCB/UPDATE=(ENH\$:CREFCB)

; Size: 154 code + 0 data bytes  
; Run Time: 00:08.3  
; Elapsed Time: 00:25.9  
; Lines/CPU Min: 3830  
; Lexemes/CPU-Min: 17470  
; Memory Used: 102 pages  
; Compilation Complete



0164

AH-BT13A-SE  
 VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY